第四屆人工智慧與應用研討會，中華民國八十八年十一月十九日，彰化大葉大學，160—166 頁。

使用決策樹來抽取文件自動分類系統中之分類規則
Extracting Classification Rules in Automatic Document Classification Systems by Using Decision Trees

洪文斌 (Wen-Bing Horng)
淡江大學 資訊工程學系
臺北縣淡水鎮英專路一五一號
TEL: (02) 2621-5656 轉 2738
FAX: (02) 2620-9749
E-mail: horng@cs.tku.edu.tw

摘要

自從 Maron 於 1961 年提出首篇文件自動分類的論文以來，傳統的分類方法不外乎機率模式與向量模式。近年來的研究也加入了統計分析、專家系統、自然語言處理、和類神經網路等先進的技術，以提高分類的正確性。以上所提的諸方法中，其對文件自動分類而言，均可視為是黑箱作業，因其分類行為或分類規則無從得知。本研究利用機械學習技術中之 Quinlan 的 C4.5 決策樹(decision trees)來抽取文件自動分類系統中之分類規則，期使文件自動分類系統之分類行為透明化，而人們可藉由所抽取之分類規則進一步來驗證文件自動分類之正確性。在本研究中，我們採用 *ACM Computing Reviews* 的分類法作為分類的依據。我們從該期刊共收錄了 56 個中類別，6424 篇論文為實驗用資料。再以其中的論文題目和出處當作該文件的素描(profile)。取其中十分之一作為測試資料，其餘為訓練資料。我們從訓練資料中，使用 Quinlan 的決策樹共抽取出 1162 條分類規則。再利用此分類規則分別對訓練文件及測試文件做分類，實驗結果分別為：訓練資料召回率為 67.7%，測試資料為 45.5%。若將上述規則再精簡成 290 條分類規則，則訓練資料召回率變為 52.3%，而測試資料略降為 43.0%。

關鍵詞：文件自動分類，決策樹，資訊檢索，機械學習

Abstract

Since Maron proposed the first paper on automatic document classification in 1961, traditionally there are two approaches used: the probability model and the vector space model. Recent research also includes the advanced techniques of statistics, expert systems, natural languages processing, and artificial neural networks to enhance the correctness of document classification. However, all of the aforementioned methods could be regarded as black boxes for automatic document classification, because there are no ways to obtain their classification behaviors or classification rules. This paper uses Quinlan's C4.5 decision trees of machine learning techniques to extract classification rules from automatic documents classification systems. In this research, the classification system of *ACM Computing Reviews* is based on. Totally 6424 papers, including 56 classes, are collected from it. The title and its source of each paper are used as its document profile. Among the collected papers, 10% of them are used as test data, and the remaining are used as training data. Totally, there are 1162 classification rules extracted from the training data using Quinlan's decision trees. These extracted classification rules are then used to categorize the training documents and test documents, respectively. The experiment results show that, the recall rates of training data and test data are 67.7% and 45.5%, respectively. If the above rules are further simplified into 290 classification rules, the recall rates of training data and test data become 52.3% and 43.0%, respectively.

Keywords: automatic document classification, decision tree, information retrieval, machine learning

1. 緒言

隨著資訊時代的來臨，電腦網路的發達，資訊正以等比級數般的數量在激增。要在這龐大的資料中，找尋相關的資訊，確非易事。因此，文件自動分類的研究便應運而生。文件自動分類的目的即是利用電腦精確及快速的計算能力，依照

某種數學模式將性質相近的的資料或文件聚集在一起，以提高文件分類的正確性與一致性，便於使用者能夠快速地檢索到相關的資訊。

Maron [11] 於 1961 年發表的論文，應該是文件自動分類領域中最早的文獻。Maron 認為，對於所要分類的文件，我們可以從文件中的某些詞找到分類的線索，稱之為關鍵詞(keywords)。若電腦也能從文件中自動找出這些關鍵詞，那麼便可以做到所謂的自動分類。在該論文中，他首先挑選了 405 篇文件,其中的 260 篇是訓練資料，另外的 145 篇是測試資料。每篇均取其摘要當作文件的素描。結果，在所有訓練資料中，共得到 3263 個不同的詞。其次，做關鍵詞篩選，把這些詞中去掉頻率最高的 55 個，及只出現一次或兩次的詞，便剩下 1088 個詞。再根據 Entropy 公式來計算，看這些詞在文件的分佈情形。只有分佈不均勻的才有分類的價值，所以把均勻者去掉。最後就只剩 90 個詞，也就是關鍵詞。他採用機率模式來作文件分類的實驗。結果顯示，在扣除不含關鍵詞及只含一個關鍵詞的文件後，訓練資料中有 84.6%的召回率(或正確率，即系統辨識正確之文件數和文件總數之比率)，而測試資料的召回率也達到 51.8%。

之後，陸續還有許多學者提出不同的作法，像 Borko 和 Bernick [3] 延續了 Maron 的實驗，嘗試用向量模式來做分類。Kar 和 White [8] 的實驗，提出了第二選擇類別來提高正確率及循序的演算法來節省時間及空間。Kwok [9] 的分類實驗，除了論文題目及摘要外，他還利用論文所引用參考文獻的題目作為分類之用。以及 Hamill 和 Zamora [4] 的分類實驗(以下簡稱為 Hamill 方法)，提出了只用文件的題目來做分類。近年來的研究也加入了統計分析、專家系統、自然語言處理、和類神經網路等先進的技術，以提高分類的正確性 [1][2][5][7][10][12][15]。然而，上述諸方法中，無論是傳統的機率與向量模式或是先進的類神經網路模式，其對文件自動分類而言，均可視為是黑箱作業，因他們的分類行為或分類規則無從得知。

本論文的主要目的，是嘗試利用機械學習技術來抽取文件自動分類系統中之符號分類規則，期使文件自動分類系統之分類行為透明化，而人們可藉由所抽取之分類規則進一步來驗證文件自動分類之正確性。在本研究中，我們採用了 *ACM Computing Reviews* 的分類法作為分類的依據，以及 Quinlan 的 C4.5 [12] 決策樹來實驗抽取文件分類的規則。我們從該期刊共收錄了 56 個中類別，6424 篇論文為實驗用資料。再以其中的論文題目和出處當作該文件的素描。取其中十分之一作為測試資料，其餘為訓練資料。我們從訓練資料中，共抽取出 1162 條分類規則。再利用此分類規則分別對訓練文件及測試文件做分類，其召回率分別為 67.7% 和 45.5%。若將上述規則再精簡成 290 條分類規則，訓練資料和測試資料的召回率亦可達到 52.3% 和 43.0%。

本論文結構如下：第二節簡介文件自動分類中不同模式的基本原理。第三節詳述本論文的實驗方法及步驟。第四節為實驗結果與討論。

2. 背景知識

本論文的主要目的，是嘗試利用 Quinlan 的 C4.5 決策樹來做文件自動分類的研究，並以傳統的機率模式、向量模式、Hamill 方法、以及倒傳遞類神經網路作為比較的標準。下面，我們簡單地介紹這些不同模式的基本原理。

假設我們從訓練文件中，共選出了 m 個關鍵詞，分別為 $T_1, T_2, \ldots, T_m$，而文件共有 n 個類別，即 $C_1, C_2, \ldots, C_n$。

2.1 機率模式

Maron [11] 的分類實驗即採用機率模式。若文件 D_i 的素描中出現了 r 個關鍵詞 $K_1, K_2, \ldots, K_r$，則此文件屬於類別 C_j 的機率為

$$P(C_j \mid K_1, K_2, \ldots, K_r)$$

根據貝氏定理(Bayesian theorem)，上式等於

$$P(C_j) \times P(K_1, K_2, \ldots, K_r \mid C_j) / P(K_1, K_2, \ldots, K_r)$$
$$= c \times P(C_j) \times P(K_1, K_2, \ldots, K_r \mid C_j)$$
$$= c \times P(C_j) \times P(K_1 \mid C_j) \times P(K_2 \mid C_j, K_1) \times \cdots$$
$$\times P(K_r \mid C_j, K_1, K_2, \ldots, K_{r-1})$$

其中 $c = 1/P(K_1, K_2, \ldots, K_r)$ 為一常數。假設關鍵詞 $K_1, K_2, \ldots, K_r$ 彼此兩兩相互獨立(mutually independent)，則上式可化減為

$$c \times P(C_j) \times P(K_1 \mid C_j) \times P(K_2 \mid C_j) \times \cdots$$
$$\times P(K_r \mid C_j)$$

因此，求出文件 D_i 在各個類別 C_j 的條件機率，取其最大值，即視為是該類別。

2.2 向量模式

Borko 和 Bernick [3] 的實驗即採用向量模式。在此模式中，類別 C_j 可用向量

$$X_j = (x_{j1}, x_{j2}, \ldots, x_{jm})$$

來表示，而文件 D_i 的素描可用向量

$$Y_i = (y_{i1}, y_{i2}, \ldots, y_{im})$$

來表示。將文件 D_i 的向量 Y_i 分別與各個類別 C_j 的向量 X_j 做內積運算

$$Y_i \bullet X_j = y_{i1}x_{j1} + y_{i2}x_{j2} + \cdots + y_{im}x_{jm}$$

亦即求出向量 Y_i 在向量 X_j 的投影量，取最大值，即為其類別。

在此實驗中，類別 C_j 向量採用原始分類比重，其定義如下：若關鍵詞 K_k 在類別 C_j 中的分類比重為 x_{jk}，則

$$x_{jk} = (d_{jk} / t_j) / (\Sigma_{l=1,n} (d_{lk} / t_l))$$

其中，d_{jk} 為關鍵詞 K_k 在類別 C_j 中出現的次數，而 t_j 為訓練資料中屬於類別 C_j 的文件總數。而代表文件 D_i 的為一二元向量，其中 $y_{ik} = 1$ 假如關鍵詞 K_k 出現在文件 D_i 中，否則為 0。

2.3 Hamill 方法

Heaps [5]曾指出，在許多不同自動分類系統中，假設關鍵詞的獨立性是難以獲致合理的結論。相反的，他建議利用主題相關性的預測。文件 D_i 對類別 C_j 的相關值 r_{ij} 可用下面公式形式表示

$$r_{ij} = f_j(\{K_{ik}\})$$

其中，$\{K_{ik}\}$ 為文件 D_i 的素描中出現的關鍵詞所成的集合。假設訓練文件 D_i 都由人工賦予對每一類別 C_j 的相關值 s_{ij}，那麼我們可以發展出下面公式形式

$$r_{ij} = f_j(\{K_{ik}\}, \{D_i\}, \{s_{ij}\})$$

使得 r_{ij} 的值最接近 s_{ij} 的值。如此，則相同公式便可用來對測試文件作分類。Heaps 建議使用如下之線性函數

$$r_{ij} = \sum_k a_{jk} w_k$$

使 r_{ij} 和 s_{ij} 的平方平均差最小化，其中，a_{jk} 為線性處理的參數，而 w_k 的值為 1 或 0，則視關鍵詞 K_{ik} 是否出現於文件 D_i 中而定。

Hamill [4] 使用條件機率 $P(C_j | K_k)$ 當成 $a_{jk} w_k$ 的值，因為 $P(C_j | K_k) = 0$ 假如關鍵詞 K_k 不存在於文件 D_i 中或不出現在類別 C_j 中。因此，$P(C_j | K_k)$ 可用來當成 w_k 假如 $w_k = 0$。對於測試文件的 n 個類別 $C_1, C_2, \ldots, C_n$，條件機率 $P(C_j | K_k)$ 可由下式計算而得

$$P(C_j | K_k) = d_{jk} / (\Sigma_{l=1,n} d_{lk})$$

若文件 D_i 的素描中包含了 r 個關鍵詞 $K_1, K_2, \ldots, K_r$，其中，d_{jk} 為關鍵詞 K_k 在類別 C_j 中出現的次數。則計算文件 D_i 對於每一類別 C_j 的相關值

$$r_{ij} = (1/r) \sum_{k=1,r} P(C_j | K_k)$$

選取其中最高的相關值 r_{ij}，則文件 D_i 應屬於類別 C_j。

2.4 倒傳遞類神經網路

倒傳遞神經網路模式是目前類神經網路學習模式中最具代表性，應用最普遍的模式。在 1985 年由 Rumelhart，Hinton，和 Williams[14] 所發表的倒傳遞神經網路論文最廣為人知。此網路模式基本上是改進了 Rosenblatt 於 1957 年所提出之感知機(Perceptron)學習理論中，無隱藏層學習演算法的缺點。此網路模式中可增加隱藏層，使網路可表現輸入處理單元間的交互影響，並且改用平滑可微分的轉換函數，使得網路可用最陡坡降法導出修正網路連結加權值的公式。其網路架構包含一輸入層，一輸出層，和數層隱藏層。在文件分類應用上，輸入層為關鍵詞，輸出層代表類別，而隱藏層則用以表示關鍵詞和類別間的交互影響關係。在學習過程中，可將輸出層的誤差值逐層倒傳回來，以修正各層間的網路連結加權值。此網路的優點為學習精度高，可處理複雜的樣本識別問題；且回想速度快。缺點則為學習速度慢，有局部最小值的問題。

2.5 Quinlan 的 C4.5 決策樹

Quinlan 的 C4.5 決策樹 [13] 是傳統符號機械學習方法中，最具代表性與實用性。它從大量的訓練文件中，利用歸納推論建構而成。在建構決策樹的過程中，不斷地應用資訊理論中之 Entropy 公式來選取適當的關鍵詞（即屬性），再以其關鍵詞作為決策樹的分枝，將原先的輸入訓練文件依其關鍵詞的存在與否歸入相對的分枝內，如是重複地選取適當的關鍵詞，再分枝，直至該分枝內所含的文件均屬於同一類別為止。當決策樹建好之後，我們可從樹根開始往各樹葉走，如是便可得出一組決策樹的分類規則，以所經過的關鍵詞為分類規則的條件，而該樹葉的類別為規則的結論。再經過一些簡化過程以去除不必要的條件，便可得到更精簡的分類規則。於是，決策樹的分類行為變成透明化，而我們可進一步去檢視所得的分類規則。我們再利用這些分類規則對原先的訓練文件和測試文件作分類，以得出其分類的正確率。

3. 實驗方法及步驟

本實驗可分為下列四個主要步驟來實施：(一)選定分類系統及實驗素材，(二)選取分類用關鍵詞，(三)抽取文件自動分類的分類規則，和(四)實驗傳統模式以為比較。以下就上述四個主要步驟做詳細說明。

3.1 選定分類系統及實驗素材

在本研究中，我們採用 *ACM Computing Reviews* 的分類系統作為分類的依據。其分類系統共有 11 個大類和 80 個中類。我們從該期刊上，收錄了自西元 1986 年 1 月份起至 1997 年 6 月份止，共 67 個中類別，6507 篇論文(詳如表一)。其中有 11 個中類別所含之論文數少於 10 篇，沒有足夠資訊以為訓練，將之刪除，並將有重覆出現之論文 49 篇去除，剩餘 56 個中類別，6424 篇論文為實驗用資料。再以其中的論文題目和出處當作該文件的素描。依論文收錄之順序，每第十篇取為測試資料，計有 643 篇文件；其餘有 5781 篇為訓練資料。

所蒐集的每篇論文，以四行格式儲存。第一行儲存論文原始編號，第二行儲存論文所屬類別，第三行儲存論文題目，第四行儲存論文出處。以下例說明：

表一：*ACM Computing Reviews* 的分類系統共有 11 個大類和 80 個中類

A. General Literature	**G. Mathematics of Computing**
A.0 General (3)*	G.0 General (0)*
A.1 Introductory and Survey (2)*	G.1 Numerical Analysis (390)
A.2 Reference (1)*	G.2 Discrete Mathematics (201)
A.m Miscellaneous (0)*	G.3 Probability and Statistics (48)
B. Hardware	G.4 Mathematical Software (29)
B.0 General (0)*	G.m Miscellaneous (16)
B.1 Control Structures and Microprogramming (13)	**H. Information Systems**
B.2 Arithmetic and Logic Structures (6)*	H.0 General (2)*
B.3 Memory Structures (42)	H.1 Models and Principles (101)
B.4 Input/Output and Data Communications (24)	H.2 Database Management (413)
B.5 Register-Transfer-Level Implementation (15)	H.3 Information Storage and Retrieval (184)
B.6 Logic Design (39)	H.4 Information Systems Applications (85)
B.7 Integrated Circuits (99)	H.5 Information Interfaces and Presentation (89)
B.m Miscellaneous (1)*	H.m Miscellaneous (0)*
C. Computer Systems Organization	**I. Computing Methodologies**
C.0 General (13)	I.0 General (0)*
C.1 Processor Architectures (124)	I.1 Algebraic Manipulation (89)
C.2 Computer-Communication Networks (275)	I.2 Artificial Intelligence (600)
C.3 Special-Purpose and Application-Based Systems (26)	I.3 Computer Graphics (193)
C.4 Performance of Systems (105)	I.4 Image Processing (55)
C.5 Computer System Implementation (26)	I.5 Pattern Recognition (64)
C.m Miscellaneous (0)*	I.6 Simulation and Modeling (62)
D. Software	I.7 Text Processing (30)
D.0 General (9)*	I.m Miscellaneous (0)*
D.1 Programming Techniques (95)	**J. Computer Applications**
D.2 Software Engineering (447)	J.0 General (1)*
D.3 Programming Languages (413)	J.1 Administrative Data Processing (22)
D.4 Operating Systems (296)	J.2 Physical Sciences and Engineering (27)
D.m Miscellaneous (17)	J.3 Life and Medical Sciences (42)
E. Data	J.4 Social and Behavioral Sciences (5)*
E.0 General (0)*	J.5 Arts and Humanities (30)
E.1 Data Structures (58)	J.6 Computer-Aided Engineering (29)
E.2 Data Storage Representations (13)	J.7 Computers in Other Systems (21)
E.3 Data Encryption (37)	J.m Miscellaneous (3)*
E.4 Coding and Information Theory (31)	**K. Computing Milieux**
E.5 Files (15)	K.0 General (0)*
E.m Miscellaneous (0)*	K.1 The Computer Industry (33)
F. Theory of Computation	K.2 History of Computing (45)
F.0 General (0)*	K.3 Computers and Education (181)
F.1 Computation by Abstract Devices (200)	K.4 Computers and Society (71)
F.2 Analysis of Alforithms and Problem Complexity (313)	K.5 Legal Aspects of Computing (16)
F.3 Logic and Meanings of Programs (185)	K.6 Management of Computing and Information Systems (106)
F.4 Mathematical Logic and Formal Languages (256)	K.7 The Computing Profession (24)
F.m Miscellaneous (0)*	K.8 Personal Computing (1)*
	K.m Miscellaneous (0)*

註：括弧內之數字為所蒐集之文件中，屬於該類別之文件總數。
標有星號“*”的類別因其所含文件數未超過 10 篇，故以捨棄。

```
@8601-0065
#I.2 H.2 H.3
$Portability of syntax and semantics in DATALOG
%ACM Trans. Office Inf. Syst.
```

第一行以@前導，其後之 8601-0065 為文件在 *ACM Computing Review* 中之原始編號，意為 1986 年一月份，流水序列為 0065 之論文。第二行以#前導，其後之 I.2 H.2 H.3 為該論文之類別，I.2 為主類別，其後為次類別。本實驗中，只採用主類別為文件分類之用。第三行以$前導，其後之 Portability of syntax and semantics in DATALOG 即為論文之題目。在輸入文件題目時，我們依原始題目之大小寫不同而將其保留，不做任何變更，如 Portability 和 DATALOG 等字。第四行以%前導，其後之 ACM Trans. Office Inf. Syst. 為論文之出處。

3.2 選取分類用關鍵詞

此為前處理步驟，旨在從訓練文件中選取具有分類價值的關鍵詞，以為分類實驗之用。在本

實驗中，我們將文件出處視為一個英文單字（或詞）。在關鍵詞選取上，我們依下列步驟取得：

一、在輸入資料時，我們觀察到同一詞(或詞組)有不同寫法，如表二所示，左邊及右邊的詞(組)均同時出現在訓練文件中，我們一律以右邊詞(組)取代。

表二：取代詞(組)對照表

取代前詞(組)	取代後詞(組)
anti-aliased	antialiased
common sense	commonsense
common-sense	commonsense
data base	database
data-base	database
data-compression	data compression
data flow	dataflow
data-flow	dataflow
fault-tolerant	fault tolerant
field-programmable	field programmable
fixed point	fixpoint
look-ahead	lookahead
multi-hop	multihop
natural-language	natural language
non-monotonic	nonmonotonic
pseudo-random	pseudorandom
random-access	random access
real time	realtime
real-time	realtime
root find	rootfind
spread sheet	spreadsheet
trade-off	tradeoff

二、因在輸入時，有大小寫區別，有些專有名詞有不同表示法，如表三所示，我們一律以右邊專有名詞取代。另外，題目第一個單字若非專有名詞，將其第一個大寫字母改為小寫。

表三：專有名詞取代對照表

取代前專有名詞	取代後專有名詞
ADA	Ada
APL*PLUS	APL*Plus
FORTRAN	Fortran
LISP	Lisp
Modula-2	Modula_2
MODULA-2	Modula_2
PROLOG	Prolog
gcd	GCD

三、將訓練資料中，以 O()，o()，或 theta() 等起頭之公式，視為演算法中之複雜度，一律以 complexity 取代。另外，1900 至 2000 的數字視為西元紀年，以 year 取代，其他數字全部刪除。因將文件出處視為一英文單字，將其以一個字串 R*nnn* 取代，*nnn* 為一流水編號。

四、將訓練文件所有題目中所出現之英文單字(即詞)，去除 235 個 stop words (如 about, but, in, not, …) 及只包含一個字元的單字，剩下的蒐集成一詞典(lexicon)。並將所有英文單字中，名詞單複數，及動詞單複數和其現在分詞、過去式、過去分詞列出其原型，以方便對照取出。這些單字原型，即構成關鍵候選詞

五、在關鍵詞的選取中，我們從訓練資料中，選出至少出現 5 次以上的關鍵候選詞，並且其 Entropy 值小於等於 $\log_{10}(20)$，共選出了 1146 個關鍵詞。其中，關鍵詞 K_k 的 Entropy 值 H_k 計算如下：

$$H_k = -\sum_{j=1,n} P(C_j \mid K_k) \times \log_2 P(C_j \mid K_k)$$

C_j 為類別。在本實驗中，共有 56 個類別，故 n 值為 56。在原先訓練文件中，有 26 篇沒有出現任何關鍵詞，去除後剩下 5755 篇；相同的，原先測試文件中，有 6 篇沒有出現任何關鍵詞，去之剩下 637 篇，此為以下實驗用之資料。

3.3 抽取文件自動分類的分類規則

在 Quinlan 的 C4.5 決策樹的文件自動分類實驗上，我們將所選出的 1146 個關鍵詞當成每篇文件的屬性，若該關鍵詞有出現在文件中，則其屬性值為 Yes；若否，則為 No。我們使用 C4.5 決策樹程式[12]，實驗了文件自動分類的規則抽取。我們先用訓練文件產生出 1162 條分類規則，再利用此分類規則分別對訓練文件和測試文件做分類，在 5755 篇訓練文件中，共有 3898 篇分類正確，其召回率為 67.7%，而在 637 篇測試資料中，共有 292 篇正確，故其召回率為 45.5%。

然而，經由以上所產生之決策樹，因其屬性（即關鍵詞）非常多，其所建立之決策樹亦將非常深，故所抽取出之分類規則，其條件部分，可能有數十個屬性以上所構成，過於複雜而難以瞭解。因此，利用 C4.5 的刪減(prune)決策樹方法，再將以上規則進一步精簡為 290 條規則，取其中數條規則列舉於下，以供參考：

Rule 19: music = Yes → class J.5 (Arts and Humanities)
Rule 26: wafer = Yes → class B.7 (Integrated Circuits)
Rule 30: curriculum = Yes → class K.3 (Computers and Education)
Rule 72: R055 = Yes, vision = Yes → class I.2 (Artificial Intelligence)
Rule 84: image = Yes, synthesis = Yes → class I.3 (Computer Graphics)
Rule 204: extract = Yes, information = Yes → class H.3 (Information Storage and Retrieval)
Rule 269: debugger = Yes → class D.2 (Software Engineering)

其中，第 72 條規則中之 R055 為文件之期刊出處：

Computer Vision, Graphics, and Image Processing。其義為：若文件出處為上述期刊並且在文件題目中出現 vision 這個關鍵詞，則該文件應分類為 I.2，即人工智慧類別(Artificial Intelligence)。若以精簡的 290 條分類規則來分類，則 5755 篇的訓練資料中，共有 3010 篇分類正確，其召回率為 52.3%，而 637 篇的測試資料中，有 274 篇正確，召回率為為 43.0%。

3.4 實驗傳統文件分類模式

此外，我們並用相同的實驗資料，實驗了傳統的機率模式、向量模式、Hamill 的分類方法、以及倒傳遞神經網路以為比較。其基本原理及作法如第二節所述。在機率模式方面，訓練資料召回率為 82.7%，測試資料為 39.4%。在向量模式方面，訓練資料召回率為 55.4%，測試資料為 43.0%。在 Hamill 方法實驗中，訓練資料召回率為 63.6%，測試資料為 55.3%。在倒傳遞網路實驗中，訓練資料召回率為 70.7%，測試資料為 57.1%。我們將以上實驗結果表列於表四，以供比較。

表四：各種文件分類實驗方法的召回率

文件分類實驗方法	訓練資料	測試資料
Quinlan's C4.5 (原始)	67.7%	45.5%
Quinlan's C4.5 (精簡)	52.3%	43.0%
倒傳遞網路	70.7%	57.1%
機率模式	82.7%	39.4%
向量模式	55.4%	43.0%
Hamill 方法	63.5%	55.3%

4. 結果與討論

在本實驗中，我們只採用了論文的題目和出處當作文件的素描，主要有兩個目的：一是以極有限的論文題目這個資訊來作文件自動分類，並抽取其分類規則；二是要和 Hamill 的實驗方法相比較，因為他們的實驗只採用了論文題目。加入論文出處主要是這項資訊明顯地提升了分類的正確率，這可從下列數據得知：以 Hamill 方法為例，當我們的實驗資料不加入論文出處時，訓練資料召回率只有 56.3%，測試資料為 47.3%。(在 Hamill 的原本論文中[4]，以論文題目為文件素描時，測試資料僅有 45.0%的召回率。)當加入論文出處時，訓練資料召回率明顯提升為 63.5%，測試資料為 55.3%。因此，論文出處是項重要的分類特徵，它可大幅提升文件辨識正確率約達 8%。故本研究除了文件題目外，也將其出處當作文件的素描。

在一般機械學習分類過程中，可分為兩個階段：即學習（或訓練）階段與測試階段。在學習階段中，依不同分類模式，利用訓練資料以求得一些分類之資訊，作為在測試階段中，利用測試資料來驗證所用分類模式的可靠性。然而，不同之分類模式，其求得分類資訊之複雜性亦不一，故在訓練階段中所需之時間亦不一。在機率模式、向量模式與 Hamill 方法中，其訓練階段主要為求得一些統計資訊，故所需時間略同，約為數分鐘。在倒傳遞網路模式中，因學習速度較慢以及需要不斷調整參數，如隱藏層節點個數等，以求得較佳之分類結果，故所需時間較長，約為數日之間。然而在 Quinlan 的 C4.5 軟體的學習階段中，因需將測試資料依其所有屬性計算 Entropy 值，作為決策樹分枝之依據，故所需計算量較大，約為四小時。將所得之分類規則再進一步簡化，則需兩日時間。（以上所做實驗，皆以 32M RAM, Pentium-100 PC 為例。）在測試階段中，各種分類模式所需時間則略同，且非常快速，約為數秒鐘，因為只要將所得的分類資訊與測試資料作一簡單運算即得。

在本實驗中，只收錄了 6424 篇論文題目作為實驗資料，可能因資料量尚不足夠，論文題目所含資訊過少，以及各個類別的文件數相差懸殊（詳見表一），而導致各種分類方法的召回率並不很高。就論文題目所含資訊過少來討論，在此研究中，實驗用文件素描只包含論文題目及其出處，平均每一篇文件只包含了 4.8 個關鍵詞，以倒傳遞網路實驗為例，辨識正確及錯誤的文件中其所包含關鍵詞數，發現在訓練資料中辨識正確的文件其所含關鍵詞數約為 5.0 個，而錯誤文件關鍵詞數約為 4.1 個；在測試資料中辨識正確的文件其所含關鍵詞數約為 5.0 個，而錯誤文件關鍵詞數約為 4.4 個。由此觀察，文件的關鍵詞數越多，越有助於辨識之正確性。

另外，由於所蒐集的 6424 篇文件，並非平均分佈在 56 個類別中。有些類別所含文件高達三、四百篇以上，有些類別僅含一、二十篇。在倒傳遞網路的實驗中，類別中所含文件數少於三十篇者，幾乎無法正確辨識，這可能是因為類別的文件數相差太懸殊，以致於在訓練中，連結加權值的修正傾向於含文件較多之類別，以是之故，含較少文件數之類別辨識率相對就要減低許多。

在機率模式中，主要以貝氏定理為分類依據，故訓練資料召回率可高達 82.7%，然而因樣本數不足，故測試資料召回率僅達 39.4%。在向量模式中，有多種方式來計算各類別在向量空間中的代表向量，此實驗僅採用簡單的原始分類比重方式來計算類別向量，所得召回率亦差強人意。然而此二方法原先是對含較多資訊的論文摘要而設計的，故在實驗中，測試結果表現較差。而 Hamill 方法即是針對含較少資訊的論文題目而設計的，故其總體召回率較佳。在倒傳遞網路實驗上，無

論是訓練資料或是測試資料，其召回率均相當優異，因為當網路結構設計的好的話，在訓練階段可得到較好的一般性(generalization)，故在測試資料召回率可高達 57.1%，為所有實驗中最高者。雖然，倒傳遞網路模式表現最佳，然而由於其為黑箱行為，其分類法則無從得知。故本研究嘗試使用 Quinlan 的 C4.5 決策樹的方法來萃取文件規則，期使文件分類行為透明化。在測試資料方面雖不及倒傳遞網路及 Hamill 方法，然而，它卻可用 290 條已精簡過的分類規則達到 43.0%的召回率，此分類規則已將文件分類的方式完全透明的表達出來，便於人類專家閱讀與驗證其正確性。

最後一點要說明的是，在我們觀察文件題目時，發現文件中出現很多英文字頭語(acronyms)，和其全文同時並存於訓練資料中，如 AI 代表 artificial intelligence，IR 代表 information retrieval 等。到目前為止，似乎沒有人探討這些字頭語對文件自動分類辨識的影響，我們希望在下一個實驗中，能對此一問題作深入之探討。

誌謝

本論文獲得國科會八十七年度計畫補助，計畫編號為 NSC 87-2213-E-032-016。另外，感謝審查委員所提供的寶貴意見,使本論文更具可讀性。在此，一併致上最深的感謝。

參考文獻

[1] 洪文斌和黃連進，“使用類神經網路來作文件自動分類之研究”，*1998 分散式系統技術及應用研討會*，台南，成功大學，民國八十七年五月，209-216 頁。

[2] M.J. Blosseville, M.J. Hebrail, M.G. Monteil, and N. Penot, “Automatic Document Classification: Natural Language Processing, Statical Analysis, and Expert System Techniques Used Together,” in *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Copenhagen, Denmark, June 21-24, 1992, pp. 51-58.

[3] H. Borko and M. Bernick, “Automatic Document Classification,” *Journal of the ACM*, Vol. 10, No. 1, 1963, pp. 151-162.

[4] K.A. Hamill and A. Zamora, “The Use of Titles for Automatic Document Classification,” *Journal of the American Society for Information Science*, Vol. 31, November 1980, pp. 396-402.

[5] H.S. Heaps, “A Theory of Relevance for Automatic Document Classification,” Information and Control, Vol. 22, No. 3, 1973, pp. 268–278.

[6] P.S. Jacobs, “Joining Statistics with NLP for Text Categorization,” in *Proceedings of the 3rd Conference on Applied Natural Language Processing*}, Trento, Italy, March 31-April 3, 1992, pp. 178-185.

[7] P.S. Jacobs, “Using Statistical Methods to Improve Knowledge-Based News Categorization,” *IEEE Expert*, Vol. 8, No. 2, April 1993, pp. 13-23.

[8] G. Kar and L.J. White, “A Distance Measure for Automatic Document Classification by Sequential Analysis,” *Information Processing and Management*, Vol. 14, 1978, pp. 57-69.

[9] K.L. Kwok, “The Use of Title and Cited Titles as Document Representation for Automatic Classification,” *Information and Management*, Vol. 11, 1975, pp. 201-206.

[10] K.J. MacLeod and W. Robertson, “A Neural Algorithm for Document Clustering,” *Information Processing and Management*, Vol. 27, No. 4, 1991, pp. 337-346.

[11] M.E. Maron, “Automatic Indexing: An Experimental Inquiry,” *Journal of the ACM*, Vol. 8, 1961, pp. 404-417.

[12] H.T. Ng, W.B. Goh, and K.L. Low, “Feature Selection, Perceptron Learning, and a Usability Case Study for Text Categorization”, in *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, 1997, pp. 67–73.

[13] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, 1993.

[14] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, “Learning Internal Representation by Error Propagation,” in *Parallel Distributed Processing*, Vol. 1, MIT Press, 1986, pp. 318-362.

[15] Y. Yang, “Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval,” in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Ireland, 1994, pp. 13-22.